UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/044,927	01/11/2002	Robert N. Goldberg	16159.018001; P6405	4976

32615 7590 01/25/2005

OSHA & MAY L.L.P./SUN
1221 MCKINNEY, SUITE 2800
HOUSTON, TX 77010

EXAMINER	
LY, ANH	

ART UNIT	PAPER NUMBER
2162	

DATE MAILED: 01/25/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application N .	Applicant(s)	
	10/044,927	GOLDBERG ET AL.	
	Examin r	Art Unit	
	Anh Ly	2162	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *16 August 2004*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-23* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-23* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. This Office Action is response to Applicants' response filed on 08/16/2004.

2. Claims 1-23 are pending in this application.

Response to Arguments

3. Applicant's arguments filed 08/16/2004 have been fully considered but they are not persuasive.

Applicants argued that, "Falls does not teach or suggest a projected object graph data structure, a variable usage specification or generating a projected object graph data structure using the variable usage specification. (Page 2, and Page 3, lines 17-18).

Falls et al. of 5,991,771 (hereinafter Falls) teaches collection of related objects for a database and each object has associated attributes and each attribute assumes one or more values at any given time depending the state or status of object the that time (col. 7, lines 24-30). The class schema of objects or object graph of the database is a projected graph data structure and each object in the class schema also has certain information associated with it (col. 8, lines 10-32) and a set of containment classes that identifies the classes permitted to contain instances of this class (col. 8, lines 35-40). Also the object is moving from one state to another by changing in the value based on the location and status of each replica object as the series of state of object and transmission (col. 13, lines 66-67 and col. 14, lines 1-65, and 16, lines 2-67).

Applicants argued that, "Falls does not teach or suggest transferring a projected object graph." (Page 3, lines 9-10).

Falls teaches transferring the objects to another location with the request as SyncUpdate (col. 15, lines 4-40 and col. 36, lines 20-58 and col. 37, lines 3-10).

Claim Rejections - 35 USC § 103

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

5. This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

6. Claims 1-4, 8, 10-14, 19, 21-23 are rejected under 35 U.S.C. 103(a) as being unpatentable over US Patent No. 5,991,771 issued to Falls et al. (hereinafter Falls) in view of PUB. No. US 2001/0034733 of Prompt et al. (hereinafter Prompt).

With respect to claim 1, Falls teaches generating a request for the projected graph data structure using a variable usage specification (object stored in a database can be obtained by querying via the information or definitions in the schema: col. 4, lines 45-51);

retrieving a server graph data structure using the request (database server where the collection of related objects to be stored to be retrieved from the client of the network: see fig. 2 and col. 7, lines 24-42 and col. 8, lines 10-55);

generating a projected graph data structure representation using the request, the server graph data-structure, and a schema associated with the server graph data structure (see fig. 2, and fig. 3, the object is retrieved with the information defined in the of class schema of the object over the client-server network).

Falls teaches client/server computer network for capturing information defining a schema for user by a agent or client of the network. The class schema is including a set of attribute definitions and a set of object class or class. Each class has at least one or more attributes. A collection of related objects or "object graph" is stored in a database (col. 7, lines 22-34) where the object or part of object to be retrieved as a projection action (col. 4, lines 45-51) and synchronization or replication is performed over the client-server network (figs 1-3) and a set of containment classes that identifies the classes permitted to contain instances of this class (col. 8, lines 35-40). Falls does not

clearly teach instantiating the projected graph data structure using the project graph data structure representation.

However, Prompt teaches the client of the network being enable to add an object to the data structure where objects are stored by instantiating that object (Page 2, section 0013).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Falls with the teachings of Prompt so as to enable to add the object to the data structure of the database where objects could be retrieved by instantiating the object over the network. The motivation being to have a database for storing the related objects from which the objects can be retrieved and synchronized or replicated by client, customer or user over the client-server network.

With respect to claim 2, Falls teaches synchronizing projected objects located on the client with distributed objects located on a server (see fig. 1-3 and col. 14, lines 8-44).

With respect to claim 3, Falls teaches wherein the projected graph data structure is an object graph (col. 4, lines 45-51).

With respect to claim 4, Falls teaches wherein the server graph data structure is an object graph (see fig. 1-3 and col. 8, lines 50-67 and col. 14, lines 8-44).

With respect to claim 8, Falls teaches wherein the variable usage specification comprises a list of required objects and object attributes (class schema describing the attributes' information for of each class object: col. 8, lines 8-67).

With respect to claim 10 Falls teaches generating a request for the projected graph data-structure using a usage variable specification (object stored in a database can be obtained by querying via the information or definitions in the schema: col. 4, lines 45-51);

retrieving a server graph data-structure using the request (database server where the collection of related objects to be stored to be retrieved from the client of the network: see fig. 2 and col. 7, lines 24-42 and col. 8, lines 10-55);

generating a projected graph data-structure representation using the request, the server graph data-structure, and a schema associated with the server graph data structure (see fig. 2, and fig. 3, the object is retrieved with the information defined in the of class schema of the object over the client-server network); and

synchronizing projected objects located on the client with distributed objects located on a server (see fig. 1-3 and col. 14, lines 8-44).

Falls teaches client/server computer network for capturing information defining a schema for user by a agent or client of the network. The class schema is including a set of attribute definitions and a set of object class or class. Each class has at least one or more attributes. A collection of related objects or "object graph" is stored in a database (col. 7, lines 22-34) where the object or part of object to be retrieved as a projection action (col. 4, lines 45-51) and synchronization or replication is performed over the client-server network (figs 1-3) and a set of containment classes that identifies the classes permitted to contain instances of this class (col. 8, lines 35-40). Falls does not

clearly teach instantiating the projected graph data-structure using the projected graph data-structure representation.

However, Prompt teaches the client of the network being enable to add an object to the data structure where objects are stored by instantiating that object (Page 2, section 0013).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Falls with the teachings of Prompt so as to enable to add the object to the data structure of the database where objects could be retrieved by instantiating the object over the network. The motivation being to have a database for storing the related objects from which the objects can be retrieved and synchronized or replicated by client, customer or user over the client-server network.

With respect to claim 11, Falls teaches a customer component that generates a request for a projected object graph (see fig. 1, network system 10 including several server and one or more client: col. 6, lines 44-52);

a service component that generates a service-side projected object graph representation (see fig. 2 the services for the objects from the agents: col. 7, lines 15-50 and col. 8, lines 8-67);

means for generating the request for the projected graph data-structure using a usage variable specification (object stored in a database can be obtained by querying via the information or definitions in the schema: col. 4, lines 45-51);

means for retrieving a server graph data-structure using the request (database server where the collection of related objects to be stored to be retrieved from the client of the network: see fig. 2 and col. 7, lines 24-42 and col. 8, lines 10-55);

means for generating the projected graph data-structure representation using the request, the server graph data-structure, and a schema associated with the server graph data-structure (see fig. 2, and fig. 3, the object is retrieved with the information defined in the of class schema of the object over the client-server network);

Falls teaches client/server computer network for capturing information defining a schema for user by a agent or client of the network. The class schema is including a set of attribute definitions and a set of object class or class. Each class has at least one or more attributes. A collection of related objects or "object graph" is stored in a database (col. 7, lines 22-34) where the object or part of object to be retrieved as a projection action (col. 4, lines 45-51) and synchronization or replication is performed over the client-server network (figs 1-3) and a set of containment classes that identifies the classes permitted to contain instances of this class (col. 8, lines 35-40). Falls does not clearly teach means for instantiating the projected graph data-structure using the projected graph data structure representation.

However, Prompt teaches the client of the network being enable to add an object to the data structure where objects are stored by instantiating that object (Page 2, section 0013).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Falls with the teachings of

Prompt so as to enable to add the object to the data structure of the database where objects could be retrieved by instantiating the object over the network. The motivation being to have a database for storing the related objects from which the objects can be retrieved and synchronized or replicated by client, customer or user over the client-server network.

With respect to claim 12, Falls teaches synchronizing projected objects located on the client with distributed objects located on a server (see fig. 1-3 and col. 14, lines 8-44).

With respect to claim 13, Falls teaches wherein the projected graph data structure is an object graph (col. 4, lines 45-51).

With respect to claim 14, Falls teaches wherein the server graph data structure is an object graph (see fig. 1-3 and col. 8, lines 50-67 and col. 14, lines 8-44).

With respect to claim 19, Falls teaches wherein the variable usage specification comprises a list of required objects and object attributes (class schema describing the attributes' information for of each class object: col. 8, lines 8-67).

With respect to claim 21, Falls teaches wherein the customer component and the service component communication over a network link (network connection: see fig. 2, item 52, col. 13, lines 40-45).

Claim 22 is essentially the same as claim 1 except that it is directed to an apparatus rather than a method, and is rejected for the same reason as applied to the claim 1 hereinabove.

Claim 23 is essentially the same as claim 2 except that it is directed to an apparatus rather than a method, and is rejected for the same reason as applied to the claim 2 hereinabove.

7. Claims 5-7 and 15-18 are rejected under 35 U.S.C. 103(a) as being unpatentable over US Patent No. 5,991,771 issued to Falls et al. (hereinafter Falls) in view of PUB. No. US 2001/0034733 of Prompt et al. (hereinafter Prompt) and further in view of Pub No. US 2002/0116412 of Barnes et al. (hereinafter Barnes).

With respect to claims 5-7, Falls in view of Prompt discloses a method as discussed in claim 1.

Falls teaches client/server computer network for capturing information defining a schema for user by a agent or client of the network. The class schema is including a set of attribute definitions and a set of object class or class. Each class has at least one or more attributes. A collection of related objects or "object graph" is stored in a database (col. 7, lines 22-34) where the object or part of object to be retrieved as a projection action (col. 4, lines 45-51) and synchronization or replication is performed over the client-server network (figs 1-3) and a set of containment classes that identifies the classes permitted to contain instances of this class (col. 8, lines 35-40). Falls does not clearly teach instantiating the projected graph data structure using the project graph data structure representation. Prompt teaches the client of the network being enable to add an object to the data structure where objects are stored by instantiating that object

(Page 2, section 0013). In combination, Falls and Prompt do not explicitly teach wherein the projected graph data structure representation comprises a hash table, an Extensible Mark-up Language document, and a serialized file.

However, Barnes teaches the objects are stored in the hash table (Page 5, section 0065 and 0066); JaveBean specification is a serialized file (Page 1, 0004 and section 0012); and XML document (see fig. 6A-6C).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Falls in view of Prompt with the teachings of Barnes so as to have the place where the object could be stored and manipulated by the user over the network an to enable to add the object to the data structure of the database where objects could be retrieved by instantiating the object over the network. The motivation being to have a database for storing the related objects from which the objects can be retrieved and synchronized or replicated by client, customer or user over the client-server network.

With respect to claims 15-17, Falls in view of Prompt discloses a method as discussed in claim 11.

Falls teaches client/server computer network for capturing information defining a schema for user by a agent or client of the network. The class schema is including a set of attribute definitions and a set of object class or class. Each class has at least one or more attributes. A collection of related objects or "object graph" is stored in a database (col. 7, lines 22-34) where the object or part of object to be retrieved as a projection action (col. 4, lines 45-51) and synchronization or replication is performed over the

client-server network (figs 1-3) and a set of containment classes that identifies the classes permitted to contain instances of this class (col. 8, lines 35-40). Falls does not clearly teach instantiating the projected graph data structure using the project graph data structure representation. Prompt teaches the client of the network being enable to add an object to the data structure where objects are stored by instantiating that object (Page 2, section 0013). In combination, Falls and Prompt do not explicitly teach wherein the projected graph data structure representation comprises a hash table, an Extensible Mark-up Language document, and a serialized file.

However, Barnes teaches the objects are stored in the hash table (Page 5, section 0065 and 0066); JaveBean specification is a serialized file (Page 1, 0004 and section 0012); and XML document (see fig. 6A-6C).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Falls in view of Prompt with the teachings of Barnes so as to have the place where the object could be stored and manipulated by the user over the network an to enable to add the object to the data structure of the database where objects could be retrieved by instantiating the object over the network. The motivation being to have a database for storing the related objects from which the objects can be retrieved and synchronized or replicated by client, customer or user over the client-server network.

With respect to claim 18, Falls in view of Prompt discloses a method as discussed in claim 11.

Falls teaches client/server computer network for capturing information defining a schema for user by a agent or client of the network. The class schema is including a set of attribute definitions and a set of object class or class. Each class has at least one or more attributes. A collection of related objects or "object graph" is stored in a database (col. 7, lines 22-34) where the object or part of object to be retrieved as a projection action (col. 4, lines 45-51) and synchronization or replication is performed over the client-server network (figs 1-3) and a set of containment classes that identifies the classes permitted to contain instances of this class (col. 8, lines 35-40). Falls does not clearly teach instantiating the projected graph data structure using the project graph data structure representation. Prompt teaches the client of the network being enable to add an object to the data structure where objects are stored by instantiating that object (Page 2, section 0013). In combination, Falls and Prompt do not explicitly teach wherein the request comprises a variable usage specification.

However, Barnes teaches JaveBean specification (Page 1, 0004 and section 0012).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Falls in view of Prompt with the teachings of Barnes so as to have the place where the object could be stored and manipulated by the user over the network and to enable to add the object to the data structure of the database where objects could be retrieved by instantiating the object over the network. The motivation being to have a database for storing the related

objects from which the objects can be retrieved and synchronized or replicated by client, customer or user over the client-server network.

8. Claims 9 and 20 are rejected under 35 U.S.C. 103(a) as being unpatentable over US Patent No. 5,991,771 issued to Falls et al. (hereinafter Falls) in view of PUB. No. US 2001/0034733 of Prompt et al. (hereinafter Prompt) and further in view of Patent No. 6,063,128 issued to Bentley et al. (hereinafter Bentley).

With respect to claim 9, Falls in view of Prompt discloses a method as discussed in claim 1.

Falls teaches client/server computer network for capturing information defining a schema for user by a agent or client of the network. The class schema is including a set of attribute definitions and a set of object class or class. Each class has at least one or more attributes. A collection of related objects or "object graph" is stored in a database (col. 7, lines 22-34) where the object or part of object to be retrieved as a projection action (col. 4, lines 45-51) and synchronization or replication is performed over the client-server network (figs 1-3) and a set of containment classes that identifies the classes permitted to contain instances of this class (col. 8, lines 35-40). Falls does not clearly teach instantiating the projected graph data structure using the project graph data structure representation. Prompt teaches the client of the network being enable to add an object to the data structure where objects are stored by instantiating that object

(Page 2, section 0013). In combination, Falls and Prompt do not explicitly teach wherein the server graph data structure is located in a persistent data store.

However, Bentley a portable persistent model constructing from a set of schemas and stored in one or more persistent stores of a project database (see fig. 1, col. 7, lines 48-56 and col. 11, lines 18-28).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Falls in view of Prompt with the teachings of Bentley so as to have one or more data persistent store as shown in fig. 1) and to enable to add the object to the data structure of the database where objects could be retrieved by instantiating the object over the network. The motivation being to have a database for storing the related objects from which the objects can be retrieved and synchronized or replicated by client, customer or user over the client-server network.

With respect to claim 20 Falls in view of Prompt discloses a method as discussed in claim 11.

Falls teaches client/server computer network for capturing information defining a schema for user by a agent or client of the network. The class schema is including a set of attribute definitions and a set of object class or class. Each class has at least one or more attributes. A collection of related objects or "object graph" is stored in a database (col. 7, lines 22-34) where the object or part of object to be retrieved as a projection action (col. 4, lines 45-51) and synchronization or replication is performed over the client-server network (figs 1-3) and a set of containment classes that identifies the

classes permitted to contain instances of this class (col. 8, lines 35-40). Falls does not clearly teach instantiating the projected graph data structure using the project graph data structure representation. Prompt teaches the client of the network being enable to add an object to the data structure where objects are stored by instantiating that object (Page 2, section 0013). In combination, Falls and Prompt do not explicitly teach wherein the server graph data structure is located in a persistent data store.

However, Bentley a portable persistent model constructing from a set of schemas and stored in one or more persistent stores of a project database (see fig. 1, col. 7, lines 48-56 and col. 11, lines 18-28).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Falls in view of Prompt with the teachings of Bentley so as to have one or more data persistent store as shown in fig. 1) and to enable to add the object to the data structure of the database where objects could be retrieved by instantiating the object over the network. The motivation being to have a database for storing the related objects from which the objects can be retrieved and synchronized or replicated by client, customer or user over the client-server network.

Conclusion

9. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Contact Information

10. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Anh Ly whose telephone number is (571) 272-4039 or via E-Mail: ANH.LY@USPTO.GOV **or fax to (571) 273-4039**. The examiner can normally be reached on TUESDAY – THURSDAY from 8:30 AM – 3:30 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John Breene, can be reached on (571) 272-4107 or Primary Examiner Jean Corrielus (571) 272-4032.

Any response to this action should be mailed to:

Commissioner of Patents and Trademarks

Washington, D.C. 20231

or faxed to: Central Fax Center (703) 872-9306

JEAN M. CORRIELUS
PRIMARY EXAMINER

ANH LY
JAN. 10th, 2005